Deterministic Finite Automata (DFA) Exercise

1 DFA Example

Let us build a DFA for the following language L over the alphabet $\Sigma = \{a, b\}$.

 $L = \{ w \in \Sigma^* | w \text{ has an even number of } a \text{'s and an odd number of } b \text{'s} \}.$

You will create this DFA with JFLAP. The basic idea is to think about what states this DFA could be in as it reads symbols from an input. We do not care about how many a's or b's are read as long as there are an even number of a's and an odd number of b's. Since we are interested in the respective parities of the numbers of a's and b's let us follow the steps below.

- 1. At any stage as the input has been read, keep track of the parity of the numbers of a's and b's that have been read so far.
- 2. How many different cases are possible?
- 3. Initially, when no symbol has been read, what are the parities?
- 4. As you read the next symbol, what happens to the parities?
- 5. In JFLAP, let's create a new DFA. Create an initial state q0. There are no a's and no b's. Thus an even number of a's and an even number of b's. Let's label this state even-even.
- 6. If the first symbol is an a, we have an odd number of a's and still an even number of b's. Create a new state q1 to represent this scenario. Label this state odd-even. Add a transition labeled a from q0 to q1, Add this stage your JFLA window should be as below.

○ ○ ○ JFLAP v8.0(2states.jflap)							
File	Edit	Input	Test	View	Convert	Help	×
				Autom	naton Editor]	
	0						
	U	~ ×	- /				
		\mathbf{N}			a		
		even	-even		l	odd-even	
-							
Fin	ite Stat	e Automa	aton (FS	A) = (Q,	Σ, δ, S, F)		1
Σ	= {	a					}
Tab	le Text	Size					
-					0		

7. If the symbol in state q0 is a b, you then have an even number of a's and an odd number of b's. Follow the steps as in 6. above to get the following.

○ ○ ○ JFLAP v8.0(3states.jflap)							
File	Edit	Input	Test	View	Convert	Help	×
					Automaton	Editor	
			2	×			
		~ ×					
			10		a	al	
		even	-even			odd-even	
			4				
		\	,				
		q	2				
		even	-000				
Fin	ite Stat	e Autom	aton (FS	A) = (Q,	Σ, δ, S, F)		
Σ	= {	a b					 }
Tab	le Text	t Size					
-							

- 8. Now let us look at q1, which represents an odd number of a's and an even number of b's. What happens when we read an a? Does it go to a new state?
- 9. Also, in q1, what happens when we read an b? Does it go to a new state?

- 10. Proceed with the DFA construction by adding all the required transitions at all states. Finally, since we are interested in the language L, what state (or states) should be accept states?
- 11. At this stage your DFA should be as below.

12. Now test your DFA with multiple inputs as shown below and verify that your results are correct. Why is λ rejected?

00	0			JFLAP v8.0(4states	.jflap)	
File	Input Help				ukiala Dua	
	even-ev a a a a b c a b c a b c a b c a b c a b c a b c a b c a b c c a b c c a b c c a b c c a b c c c c	a in a a a	q1 pdd-even q3 pdd-odd	Automaton Editor	Table Text Size Table Text Size Input a a b b a a b a a b b a a a a a a b b b b a b a b a b a b a b a b λ a a a b b a a a b b a a a b b b a a a b b b a a a a a b b b a a b b b b a a b a b b b a a b b b a a b b b a a b b b a a b b b a a b a b b b a a b a b b b a a b a b b b a a b a b b b a a b a b b b a a b a b b b a a b a b b b a a b a b b b a a b a b b b a a b a b b b a a b a b b b a a b a b b b a a b a b b b a a b a b b b a a b b b a a b a b b b a a b a b b b a a b a b b b a a b a b b b a a b a b b b a a b a b b b a a b a b b b a a b b b a a b b b a a b b b a a b a b b b a a b a b b b a a b a b b b a a b a b b b a a b b b a a b b b a a b a b b b a a b a b b b a a b a b b b a a b a b b b a a b b b a a b b b a a b b b a a b b b a a b b b a a b b b a a b b b a a b b b a a b b b b a a b b b a a b b b a a b b b a a b b b a a b b b b a a b b b a a b b b b a a b b b b a a b b b a a b b b a a b b b b a a b b b b a a b b b b a a b b b b a a b b b b a a b b b b a a b b b a a b b b b a a b a b b b a a b a b b b a a b a b a b a b a b b b a a b a b b b a a b a b a b a b a b a b b b a a b b b a a b b b a a b b b a a b b b a a b b b b a a b b b a a b b b a a b b b a a b b b a a b b b b a a b b b b a a b b b b a a b b b b a a b b b b a a b b b b a a b b b b a a b b b b a a b b b b a a b b b b a a b b b a a b b b a a b b b a a b b b a a b b b a a b b b a a b b b a a b b b a a b b b a a b b b a a b b b	Result Reject Accept Reject Reject Reject Reject Reject Reject Accept Accept
					Load Inputs Run Inputs Clear Enter A	View Trace

2 Additional Exercises

- 1. Repeat the above steps for the language $L_1 = \{w \in \Sigma^* | w \text{ has an odd number of } a$'s and an odd number of b's}
- 2. Repeat the above steps for the language $L_2 = \{ w \in \Sigma^* | \text{ The number of } a \text{'s in } w \text{ is a multiple of 3} \}$

3 References

- 1. Introduction to the Theory of Computation (Third Edition), Michael Sipser. Cengage Learning. 2013.
- 2. JFLAP An Interactive Formal Languages and Automata Package, Susan H. Rodger and Thomas W Finley. Jones and Bartlett Publishers. 2006